Changes to the Specification

Please rewrite the paragraph beginning at page 8, line 16 of the specification as filed (paragraph **[0028]** of corresponding US PAP 2004/0068733) to read as follows:

"Moving now to FIGS. 3a and 3b, the flow of activities of a generic resource management process 300 is illustrated. Whenever a WMI client needs to perform ~~some~~ operations on the managed objects, a corresponding method exposed by the interface "IWbemServices" of the CIMOM is called. If the operation only involves static management data that is held ~~hold~~ in the repository, the CIMOM processes the request directly. Conversely, the relevant CIM class in the repository is inspected to identify the associated WMI providers to be exploited. One ~~exploited; one~~ or more methods is ~~are~~ then called on each WMI provider for implementing the requested service."

Please rewrite the paragraph beginning on page 11, line 2 of the specification as filed (paragraph **[0033]** of corresponding US PAP 2004/0068733) to read as follows:

"With reference now to block 327, a method corresponding to the one called on the bridge provider is called on the intermediate object (using the JNI); the identifier of the managed object (namespace path and modelname), the name of the Java provider and the converted parameters are passed to this method. In response thereto, the intermediate object locks a further semaphore at block 330, in order to ensure that the operations relating to the instantiation of the Java provider are carried out in mutual exclusion (among concurrent threads running the intermediate object). A test is made at decision block 333 to determine whether the Java provider has already been instantiated; for this purpose, the intermediate object verifies whether an identifier of the Java provider is stored in a global object "Provider_table" of the type "Hashtable". If not, the process continues to block 336; the intermediate object instantiates ~~intantiates~~ the Java provider and updates the object "Provider_table" accordingly. The process then continues to block 339. Conversely (Java provider already instantiated), the flow of activities descends into block 339 directly. Considering now block 339, the semaphore on the Java provider is unlocked."